

BUNDESREPUBLIK DEUTSCHLAND

**PRIORITY
DOCUMENT**

SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)



REC'D 19 MAY 2004

WIPO

PCT

08.04.2004

Prioritätsbescheinigung über die Einreichung einer Patentanmeldung

Aktenzeichen: 103 17 650.0

Anmeldetag: 17. April 2003

Anmelder/Inhaber: ROBERT BOSCH GMBH,
70469 Stuttgart/DE

Bezeichnung: Programmgesteuerte Einheit und Verfahren

IPC: G 06 F 11/16

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

München, den 19. Februar 2004
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag

10.11.05

ROBERT BOSCH GMBH, 70442 Stuttgart

Programmgesteuerte Einheit und Verfahren

5 STAND DER TECHNIK

Die Erfindung betrifft eine programmgesteuerte Einheit sowie ein Verfahren zum Betreiben dieser programmgesteuerten Einheit.

- 10 Derartige programmgesteuerte Einheiten sind beispielsweise als Mikroprozessoren, Mikrocontroller, Signalprozessoren oder dergleichen ausgebildet. Ein Mikrocontroller weist einen Mikrocontrollerkern, den sogenannten Core, einen oder mehrere Speicher (Programmspeicher, Datenspeicher, etc.), Peripheriekomponenten (Oszillator, I/O-Ports, Timer, AD-Wandler, DA-Wandler, Kommunikationschnittstellen) und ein Interrupt-System auf, die gemeinsam auf einem Chip integriert sind und die
- 15 über einen bzw. mehrere Busse (interner, externer Daten-/Adressbus) miteinander verbunden sind. Der Aufbau und die Funktionsweise einer solchen programmgesteuerten Einheit ist vielfach bekannt, so dass darauf nicht näher eingegangen wird.

- Der Mikrocontrollerkern ist im Sinne eines modularen Mikrocontrollerkonzepts die On-Chip integrierte zentrale Steuereinheit (CPU). Dieser beinhaltet im wesentlichen ein mehr oder weniger komplexes Steuerwerk, mehrere Register (Datenregister, Adressregister), eine Bussteuereinheit und eine Recheneinheit (ALU = arithmetisch logische Einheit), die die eigentliche Daten verarbeitende Funktion übernimmt. Eine solche ALU-Recheneinheit kann meist nur einfache Elementaroperationen mit maximal zwei beteiligten Eingangsdaten (Operanden) ausführen. Diese Operanden sowie die Ergebnisse der Berechnung können vor bzw. nach der Verarbeitung in eigens dafür vorgesehenen Registern oder Speicherstellen untergebracht sein. Bei der Verarbeitung der Operanden kann es allerdings zu Fehlern kommen, die sich nachteilig auf das Ergebnis auswirken. Ein solcher Fehler kann dadurch entstehen, dass zumindest ein eingangsseitig in die ALU-Einheit eingekoppelter Operand verfälscht wird. Dies kann beispielsweise dadurch geschehen, dass das jeweilige Eingangsdatum repräsentierende Potenzial höher bzw. niedriger als vorgesehen ist. Ist diese Ladungsänderung groß genug, kann ein
- 20 einen logischen Zustand repräsentierendes Potenzial in einen anderen logischen Zustand repräsentierendes Potenzial geändert werden. Beispielsweise kann ein eine logische "1" repräsentierendes Potenzial in ein eine logische "0" repräsentierendes Potenzial und umgekehrt geändert werden, wodurch aber das resultierende Ergebnis signifikant verfälscht wird.

5

Mit der zunehmenden Entwicklung der Halbleiterprozesstechnik hin zu kleineren Abmessungen und niedrigeren Betriebsspannungen nimmt die Wahrscheinlichkeit solcher Fehler zu. Aus diesem Grunde

sind moderne Mikroprozessorsysteme mit einem System zu Fehlererkennung bzw. Fehlerbeseitigung ausgestattet, mit denen ein auftretender Fehler identifiziert und angezeigt werden kann (Failure Identification) bzw. in Abhängigkeit von der Funktionalität des Systems Vorkehrungen im Falle eines auftretenden Fehlers getroffen werden können. Ein derartiges Fehlerkorrektursystem kann beispielsweise mittels einer ECC-Fehlerkorrektur (Error Checking and Correction) ausgestattet sein, die zum Schutz wichtiger Daten beiträgt. Um auf Fehler reagieren zu können, sind moderne Mikrokontrollersysteme in der Regel mit einer Fehlererkennungssystemen basierend auf einer redundanten Systemfunktionalität ausgestattet. Eine Systemredundanz kann z.B. durch zeitlich versetzte mehrfache Berechnung (Temporary Redundancy) oder durch zusätzliche Schaltkreise (Hardware Redundancy) realisiert werden. Im ersteren Falle, in dem ein Applikationsprogramm mehrfach zeitlich hintereinander durchgeführt wird, können zwar sporadische bzw. statistische Fehler, welche während des Betriebes entstehen, erkannt werden. Allerdings erlaubt diese Art von Redundanz lediglich eine Fehlererkennung und eine eingeschränkte Fail-Safe-Funktionalität, die darüber hinaus auch sehr zeitaufwendig ist, was die Leistungsfähigkeit des gesamten Systems beeinträchtigt. Eine Fehlerbeseitigung ist hier nicht möglich.

Aus diesem Grunde werden hauptsächlich Fehlererkennungssysteme auf der Basis einer Hardware-Redundanz verwendet, bei denen die redundante, d. h. doppelt vorgesehene Hardware das Applikationsprogramm parallel ausführt. In der internationalen Patentanmeldung WO 01/46806 mit dem Titel "Firmware Mechanism for Correcting soft Errors", welches dem Deutschen Patent Nr. DE 100 85 324 T1 entspricht, ist ein Computersystem beschrieben, das eine Hardware redundante Fehlererkennung aufweist. Das in der WO 01/46806 beschriebene Computersystem besteht aus zwei unabhängig voneinander betreibbaren Mikroprozessorkernen (Cores) und einer den beiden Cores nachgeschalteten Vergleichseinheit. In den beiden Cores können in einem ersten Betriebsmodus (Normalbetrieb) Befehle und Daten unabhängig voneinander bearbeitet werden. In einem zweiten, sogenannten Lock-Step-Betriebsmodus (Testbetrieb) werden die beiden Cores redundant betrieben, d. h. in beiden Cores werden gleiche Befehle verarbeitet. Die Ergebnisse der im redundanten Betriebsmodus betriebenen Cores werden nach Maßgabe einer Fehlerbehandlungsroutine in der Vergleichseinheit miteinander verglichen und bei Nichtübereinstimmen wird ein Fehlersignal erzeugt. Auf diese Weise können die Registerinhalte der Cores gesichert werden. Aus den so gesicherten Daten kann der Status des Mikroprozessors vor dem Auftreten des Fehlerereignisses wieder hergestellt werden.

Nachteilig an der in der WO 01/46806 beschriebenen Lösung ist der dafür erforderliche Zusatzaufwand zur Bereitstellung des redundanten Systems, vor allem da dort der gesamte Core doppelt vorgesehen ist. Insbesondere bei sehr komplexen Mikrocontrollern mit folglich einem komplexen Steuerwerk und einer komplexen Bussteuereinheit ist der für die Redundanz erforderliche zusätzliche Chipflächenaufwand sehr groß. Bei Chipflächen kritischen Mikrocontrollersystemen ist die Bereitstellung solcher Chipflächen verbrauchender Einheiten kontraproduktiv und wird vom Verbraucher zuneh-

mend nicht mehr akzeptiert. Allein aus diesem Grunde besteht somit der Bedarf, durch eine Verringerung der Chipfläche und damit einer Reduzierung der Produktkosten sich von im wesentlichen funktionsgleichen Konkurrenzprodukten auf dem Markt zu unterscheiden. Dies stellt einen erheblichen Wettbewerbsvorteil dar.

5 Mit der in der WO 01/46806 beschriebenen Anordnung lässt sich außerdem keine Fehlerqualifizierung durchführen, so dass nicht festgestellt werden kann, wo der Fehler tatsächlich aufgetreten ist. Es findet lediglich eine Fehlererkennung statt. Ein Fehler kann jedoch an verschiedenen Stellen des Systems auftreten, z.B. kann ein Fehler auf einer Busleitung oder aufgrund einer fehlerhaften Operation innerhalb einer Recheneinheit bzw. einer Vergleichseinheit auftreten. Es besteht damit auch der Bedarf einer Fehlerqualifizierung.

VORTEILE DER ERFINDUNG

15 Die erfindungsgemäße programmgesteuerte Einheit mit den Merkmalen des Patenanspruchs 1 sowie das erfindungsgemäße Verfahren mit den Merkmalen des Patentanspruchs 11 weist gegenüber den obigen bekannten Lösungsansätzen den Vorteil auf, eine insbesondere hinsichtlich des Chipflächenbedarfes optimierte und vereinfachte Fehlerkorrektur bereitzustellen.

20 Der Erfindung liegt die Erkenntnis zugrunde, dass für eine Fehlererkennung nicht der gesamte Mikrocontrollerkern redundant sein muss. Es reicht vielmehr völlig aus, dass lediglich die Ausführungseinheit, in der letztendlich die Rechenoperationen vorgenommen wird, redundant ist. Eine solche programmgesteuerte Einheit mit Fehlererkennung kommt damit im Vergleich zu dem obigen bekannten System mit sehr viel weniger Chipfläche aus, da auf das doppelte Vorsehen von Steuerwerk, Bussteuereinheit und Register, die innerhalb eines Mikrocontrollerkerns die größte Chipfläche einnehmen, verzichtet werden kann.

Die der Erfindung zugrunde liegende Idee besteht somit in der Duplizierung lediglich der Ausführungseinheit des Mikrocontrollerkerns. Damit ist eine voll funktionsfähige Fehlererkennung möglich, wobei die restlichen Komponenten eines Mikrocontrollerkerns, wie zum Beispiel Steuerwerk und Bussteuereinheit, durch andere Fehlererkennungsmechanismen basierend auf Fehlererkennung- oder Fehlerkorrekturcodes abgesichert werden. Es lässt sich damit eine programmgesteuerte Einheit mit Fehlererkennungseinrichtung bereitstellen, die mit deutlich geringerer Chipfläche auskommt als herkömmliche programmgesteuerte Einheiten, die zur Fehlererkennung einen mit zwei Mikrocontrollerkernen ausgestatteten, sogenannten Dual-Core-Mikrocontroller aufweisen. Die Chipfläche der erfindungsgemäßen programmgesteuerten Einheit bzw. deren Fehlererkennungseinrichtung ist zwar größer als die Chipfläche von sogenannten Single-Core programmgesteuerten Einheiten, die also lediglich

einen Mikrocontrollerkern und somit keine Fehlererkennungseinrichtung aufweisen. Jedoch ist die Chipfläche der erfindungsgemäßen programmgesteuerten Einheit bzw. deren Fehlererkennungseinrichtung gegenüber den Dual-Core-Mikrocontrollern signifikant reduziert.

- 5 Der besondere Vorteil des erfindungsgemäßen Verfahrens bzw. der erfindungsgemäßen Anordnung besteht auch darin, dass ein Fehler innerhalb eines Taktzyklus erkannt werden kann und somit sehr schnell entsprechende Korrekturmaßnahmen eingeleitet werden können. Auf diese Weise ist die Leistungsfähigkeit des gesamten Systems nahezu unbeeinträchtigt.
- 10 Ein weiterer Vorteil der vorliegenden Erfindung besteht darin, dass neben der Erkennung eines Fehlers auch eine Fehlerqualifizierung möglich ist, dass heißt es kann bestimmt werden, an welchem Fehlerort innerhalb der programmgesteuerten Einheit der Fehler aufgetreten ist.

- 15 Vorteilhafte Ausgestaltungen und Weiterbildungen der Erfindung sind den Unteransprüchen sowie der Beschreibung unter Bezugnahme auf die Zeichnung entnehmbar.

- Die erfindungsgemäße programmgesteuerte Einheit weist einen ersten Betriebsmodus, nachfolgend als Normalbetrieb bezeichnet, und einen zweiten Betriebsmodus, nachfolgend als Testbetrieb bezeichnet, auf. Die programmgesteuerte Einheit weist einen einzigen Mikrocontrollerkern auf, der jedoch mit
- 20 zwei Ausführungseinheiten ausgestattet ist. Unter einer Ausführungseinheiten ist z. B. eine arithmetisch logische Einheit (ALU) zu verstehen, in der die eigentlichen Datenverarbeitungsfunktionen vorgenommen werden. Die Ausführungseinheit wird häufig auch als Rechenwerk oder Recheneinheit bezeichnet. Im Normalbetrieb können, müssen jedoch nicht notwendigerweise, die beiden Ausführungseinheiten Befehle parallel verarbeiten. Im Testbetrieb erfolgt die Fehlererkennung. Im Testbe-
- 25 trieb werden gleiche Befehle parallel in beide Ausführungseinheiten eingekoppelt. Aus dem Vergleich beider Ergebnisse kann somit das Vorhandensein eines Fehlers detektiert werden.

- Zu diesem Zweck ist eine Fehlererkennungseinrichtung vorgesehen, die im Testbetrieb eine Fehlererkennung und/oder eine Fehlerkorrektur vornimmt. Eine Korrektur eines in der Ausführungseinheit
- 30 entdeckten Fehlers erfolgt nach Maßgabe einer Fehlerbehandlungsroutine (Fehlerkorrekturverfahren) durch Wiederholung eines entsprechenden Befehls. Hierzu sind je nach Beschaffenheit des Cores Schattenregister für die Eingangsregister notwendig.

- Zum Zwecke der Fehlerkorrektur weist die Fehlererkennungseinrichtung einen Kodierer auf, mittels
- 5 dem Daten mit einem Fehlererkennungs- und/oder einem Fehlerkorrekturcode versehen werden. Dabei werden Ergebnisdaten, die in Folge der Berechnung ausgangsseitig an den Ausführungseinheiten abgreifbar sind, mit dem entsprechenden Fehlerkerkennungscode bzw. Fehlerkorrekturcode versehen.

Eingangsseitig in die Ausführungseinheit eingekoppelte Daten werden typischerweise nicht mit einem Fehlererkennungs- und/oder Fehlerkorrekturcode versehen. Es wird hier lediglich eine Prüfsumme der eingekoppelten Daten gebildet. Diese Prüfsumme wird mit den in den Registern abgelegten Prüfsummen verglichen und bei einer Verfälschung werden die Daten korrigiert und nochmals in die Ausführungseinheit eingekoppelt, jedoch ohne Prüfsumme.

In einer ersten Ausgestaltung weist die Fehlererkennungseinrichtung eine erste Vergleichseinheit auf, die den beiden Ausführungseinheiten ausgangsseitig nachgeschaltet ist. Diese Vergleichseinheit vergleicht die von den Recheneinheiten berechneten Ergebnisdaten bzw. deren Fehlerkorrekturcodierung nach Maßgabe einer Fehlerbehandlungsroutine. Im Falle eines erkannten Fehlers, d. h. für den Fall, dass die Ergebnisdaten bzw. die Fehlerkorrekturcodierung nicht übereinstimmen, wird dies als Fehler erkannt und ein Fehlersignal ausgegeben.

In einer weiteren Ausgestaltung weist die Fehlererkennungseinrichtung eine zweite Vergleichseinheit auf, die zumindest einer der Ausführungseinheiten eingangsseitig vorgeschaltet ist. Diese Vergleichseinheit vergleicht die einer jeweiligen Ausführungseinheit zugeführten Operanden bzw. deren Fehlerkorrekturcodierung nach Maßgabe einer Fehlerbehandlungsroutine. Bei Vorhandensein eines Fehlers, d. h. bei Abweichen der in der Vergleichseinheit miteinander verglichenen Eingangsdaten bzw. Fehlerkorrekturcodierung, wird dies als Fehler interpretiert, woraufhin ein Fehlersignal ausgegeben wird.

In einer weiteren Ausgestaltung ist ein gemeinsames Datenregister, welches im Testbetriebsmodus beiden Ausführungseinheiten zugeordnet ist, vorgesehen. In dieses gemeinsame Datenregister lassen sich Daten, die beispielsweise über einen Bus den Ausführungseinheiten zugeführt werden sollen, ablegen.

In einer weiteren Ausgestaltung kann ein Schattenregister vorgesehen sein, in dem die den jeweiligen Ausführungseinheiten im Testbetriebsmodus vor der Berechnung zuletzt zugeführten Eingangsdaten abgelegt sind. In einer sehr einfachen Ausgestaltung kann ein solches Schattenregister als einfaches FIFO ausgebildet sein. Dieses FIFO wird erst dann weitergeschaltet und kann damit wieder beschrieben werden, wenn der Vergleich innerhalb der Vergleichseinheiten ergibt, dass kein Fehler vorliegt.

Hierfür ist vorteilhafterweise eine Steuereinrichtung vorgesehen, die eingangsseitig mit der Fehlererkennungseinrichtung und ausgangsseitig mit dem Schattenregister gekoppelt ist. Erkennt die Fehlererkennungseinrichtung, dass kein Fehler vorliegt, dann wird von der Steuereinrichtung ein Freigabesignal erzeugt, welches das Schattenregister zum erneuten Beschreiben wieder freigibt.

Die erfindungsgemäße programmgesteuerte Einheit kann beispielsweise als Mikrokontroller, Mikroprozessor, Signalprozessor oder wie auch immer ausgestaltete Steuereinheit realisiert sein.

In einem sehr vorteilhaften erfindungsgemäßen Verfahren werden die Eingangsdaten bzw. die berechneten Ergebnisdaten oder deren Fehlerkodierung miteinander verglichen. Ergibt dieser Vergleich, dass die Daten bzw. Codes nicht miteinander übereinstimmen, dann wird dies als Fehler interpretiert und es wird ein Fehlersignal erzeugt.

In einer besonders vorteilhaften Ausgestaltung wird für jeden dieser Fehler ein eigenes Fehlersignal ausgegeben, so dass aus dem Fehlersignal eine Lokalisierung des Fehlerortes möglich ist. Es lassen sich damit verschiedene Fehlerarten voneinander unterscheiden. Beispielsweise kann so ein Fehler, der durch fehlerhafte Codierung auftritt, von einem Fehler, der aufgrund von über die Busleitungen eingekoppelten fehlerhaften Daten entsteht oder der innerhalb der Recheneinheit generiert wird, unterschieden werden. Damit ist in sehr vorteilhafter Weise neben einer Fehlerquantifizierung auch eine Fehlerqualifizierung möglich.

In einer besonders vorteilhaften Ausgestaltung werden die eingangsseitig in die Recheneinheiten eingekoppelten Operanden zunächst beiden Ausführungseinheiten zugeführt. Erst anschließend wird aus diesen Eingangsdaten eine Prüfsumme (z.B. Parity, CRC, ECC) gebildet und den eingangsseitigen Vergleichen zugeführt. Damit wird das Datenverarbeitungssystem durch die eingangsseitige Fehlerkorrektur in seiner Leistungsfähigkeit nicht nennenswert beeinträchtigt.

In einer Weiterbildung des erfindungsgemäßen Verfahrens werden die abgelegten Eingangsdaten der letzten Berechnung erst wieder neu beschrieben, wenn ein Vergleich innerhalb einer Fehlererkennungseinrichtung ergibt, dass kein Fehler vorliegt. Auf diese Weise wird gewährleistet, dass die ursprünglich eingekoppelten Daten bzw. deren Codierung auch bei einer fehlerhaften Berechnung in einer der Ausführungseinheiten bzw. bei einem Codierungsfehler nicht verloren gehen.

ZEICHNUNGEN

Die Erfindung wird nachfolgend anhand der in den Figuren der Zeichnung angegebenen Ausführungsbeispiele näher erläutert. Es zeigt dabei:

Figur 1 ein erstes Funktionsschaltbild, anhand dessen die erfindungsgemäße programmgesteuerte Einheit sowie deren Betrieb beschrieben wird;

Figur 2 ein zweites Funktionsschaltbild, anhand dessen die erfindungsgemäße programmgesteuerte Einheit sowie deren Betrieb näher beschrieben wird.

BESCHREIBUNG DER AUSFÜHRUNGSBEISPIELE

In den Figuren der Zeichnung sind gleiche bzw. funktionsgleiche Elemente - sofern nichts anderes angegeben ist - mit gleichen Bezugszeichen versehen worden. Die erfindungsgemäße programmgesteuerte Einheit sowie deren Komponenten wie Mikrocontrollerkern (CPU), Speichereinheiten, Peripherieeinheiten, etc. sind der besseren Übersichtlichkeit halber in den Figur 1 und 2 nicht dargestellt worden.

In den Figur 1 und 2 sind mit den Bezugszeichen 1 und 2 jeweils arithmetisch logische Einheiten (ALU) bezeichnet. Eine jeweilige ALU-Einheit 1, 2 weist zwei Eingänge und einen Ausgang auf. In einem Testbetrieb können die zur Ausführung vorgesehenen Operanden direkt vom Bus 3 in die Eingänge der ALU-Einheiten 1, 2 eingekoppelt werden (nicht dargestellt) oder vorher in einem eigens dafür vorgesehenen Operandenregister 8, 9 abgelegt sein. Diese Operandenregister 8, 9 sind direkt mit dem Datenbus 3 gekoppelt. Die beiden ALU-Einheiten 1, 2 werden also aus den gleichen Operandenregistern 8, 9 versorgt. Zusätzlich kann vorgesehen sein, dass die jeweiligen Operanden über den Bus bereits mit einer ECC-Kodierung versehen sind, welche in den Registerbereichen 8', 9' abgelegt werden.

Bei Einkopplung der jeweiligen Operanden in die ALU-Einheiten 1, 2 muss besonderer Wert auf die korrekte Dateneingabe gelegt werden. Werden z. B. die gleichen fehlerhaften Operanden in die beiden ALU-Einheiten 1, 2 eingekoppelt, ist ein Fehler am Ausgang der ALU-Einheiten 1, 2 nicht erkennbar. Es muss daher sichergestellt sein, dass zumindest eine der ALU-Einheiten 1, 2 einen korrekten Dateneingabewert erhält bzw. auch beide ALU-Einheiten 1, 2 unterschiedliche, jedoch falsche Dateneingabewerte erhalten. Dies wird dadurch sichergestellt, dass von zumindest einem Eingangswert einer ALU-Einheit 1, 2 eine Prüfsumme (z.B. Parity, CRC, ECC) gebildet wird. In einer eigens vorgesehenen Vergleichseinheit 5, 6 wird die ECC-Kodierung 10', 11' aus diesen zusätzlichen Datenregistern 10, 11 mit der ECC-Kodierung 8', 9' aus dem ursprünglichen Quellregister 8, 9 verglichen. Optional können auch die Eingangsdaten aus den Registern 10, 11 mit denen aus den Quellregistern 8, 9 verglichen werden (nicht dargestellt). Ergibt sich ein Unterschied in der ECC-Kodierung bzw. bei den Operanden, dann wird dies als Fehler interpretiert und es wird ein Fehlersignal ausgegeben.

Dieser Vergleich erfolgt vorteilhafterweise während der Bearbeitung der Operanden in den ALU-Einheiten 1, 2, so dass diese eingangsseitige Fehlererkennung und Fehlerkorrektur nahezu ohne Leistungsverlust einhergeht. Erkennt eine der Vergleichseinheiten 5, 6 einen Fehler, so kann die Berech-

nung innerhalb des nächsten Zyklus wiederholt werden. Dabei ist der Einsatz eines Schattenregisters empfehlenswert, um immer die Operanden der letzten Berechnung zu sichern, damit sie in einem Fehlerfall schnell wieder verfügbar sind. Auf die Bereitstellung eines Schattenregisters kann allerdings verzichtet werden, wenn die jeweiligen Operandenregister 10, 11 erst wieder durch ein Freigabesignal auf Grund des Nichtvorhandenseins eines Fehlers erneut beschrieben werden. Im Falle eines Fehlers liefern die Vergleichseinheiten 5, 6 ein Fehlersignal, wodurch die Operandenregister 10, 11 nicht erneut beschrieben werden.

Die ALU-Einheiten 1, 2 erzeugen ausgangsseitig jeweils ein Ergebnis. Die von den ALU-Einheiten 1, 2 bereitgestellten Ergebnisdaten bzw. deren ECC-Kodierung werden in den Ergebnisregistern 12, 13, 12', 13' abgelegt. Diese Ergebnisdaten und/oder deren Kodierung werden in der Vergleichseinheit 14 miteinander verglichen. Im Falle des Nichtvorhandenseins eines Fehlers wird ein Freigabesignal 16 erzeugt. Dieses Freigabesignal 16 wird in die Freigabeeinrichtung 15 eingekoppelt, welche dazu zu veranlasst wird, die Ergebnisdaten auf einen Bus 4 zu schreiben. Über den Bus 4 können diese Ergebnisdaten dann weiter verarbeitet werden.

Das Freigabesignal 16 kann ferner dazu genutzt werden, die Register 8 – 11 wieder freizuschalten, so dass die nächsten Operanden von dem Bus 3 ausgelesen werden können und in den ALU-Einheiten 1, 2 verarbeitet werden können.

Mit der Anordnung in Figur 1 wird das Ergebnis nicht überprüft. Hier werden lediglich die Ergebnisdaten in der Vergleichseinheit 14 miteinander verglichen. Eine Überprüfung der ECC-Kodierung der Ergebnisdaten wird erst durch die Anordnung in Figur 2 möglich, bei der sowohl die Ergebnisdaten, als auch deren ECC-Codierung miteinander in der Vergleichseinheit 14 verglichen werden.

Der in den Figuren 1 und 2 angegebenen Fehlererkennungsanordnung werden alle transienten Fehler, permanenten Fehler und auch Laufzeitfehler erkannt. Laufzeitfehler innerhalb einer ALU-Einheit 1, 2 werden erkannt, wenn das Ergebnis nicht oder zu spät zu der Vergleichseinheit 12 gelangt und somit ein Vergleich mit einem Teilergebnis erfolgt. Durch die Absicherung der Operandenregister 8, 9, 10, 11 mit Fehlererkennungs- und Fehlerkorrekturcode und dem Vergleich der Endergebnisse ist der jeweilige Fehlerort und Fehlerzeitpunkt genau zu lokalisieren. Somit kann auf eine transiente Störung sehr schnell reagiert werden.

Es ergeben sich somit die folgenden Möglichkeiten der Fehlerlokalisierung:

- Ergibt ein Vergleich der Ergebnisdaten in der Vergleichseinheit 14 einen Unterschied, dann kann auf einen Fehler innerhalb einer der ALU-Einheiten 1, 2 geschlossen werden.

- Ergibt ein Vergleich der ECC-Kodierung in einer der Vergleichseinheiten 5, 6 einen Unterschied, dann kann auf ein fehlerhaftes Signal von dem Bus 3 bzw. vorgeschalteten Komponenten geschlossen werden.
- Ergibt ein Vergleich der ECC-Kodierung in der Vergleichseinheit 14 einen Unterschied, dann kann auf eine fehlerhafte Kodierung des Ergebnisses geschlossen werden.

Obwohl die vorliegende Erfindung vorstehend anhand eines bevorzugten Ausführungsbeispiels beschreiben wurde, ist sie darauf nicht beschränkt, sondern auf vielfältige dem Fachmann bekannte Art und Weise modifizierbar.

ROBERT BOSCH GMBH, 70442 Stuttgart

Programmgesteuerte Einheit und Verfahren

5

PATENTANSPRÜCHE

- 10 1. Programmgesteuerte Einheit, mit einem einzigen Kontrollerkern (Core), der eine erste und zumindest eine zweite Ausführungseinheit (1, 2) aufweist, die in einem ersten Betriebsmodus unabhängig voneinander betreibbar sind und die in einem zweiten Betriebsmodus parallel die gleichen Befehle abarbeiten.
- 15 Programmgesteuerte Einheit nach Anspruch 1,
dadurch gekennzeichnet,
dass eine Fehlererkennungseinrichtung (5, 6, 10, 11, 14) vorgesehen ist, die in dem zweiten Betriebsmodus nach Maßgabe einer Fehlerbehandlungsroutine eine Fehlererkennung und/oder Fehlerkorrektur vornimmt.
- 20 3. Programmgesteuerte Einheit nach Anspruch 2,
dadurch gekennzeichnet,
dass die Fehlererkennungseinrichtung (5, 6, 10', 11', 13', 14) einen Kodierer (10', 11', 13') aufweist, der den Ausführungseinheiten (1, 2) eingangsseitig zugeführte Eingangsdaten und/oder von einer jeweiligen Ausführungseinheit berechnete Ausgangssignale mit einem Fehlererkennungscode und/oder mit einem Fehlerkorrekturcode versieht.
- 25 4. Programmgesteuerte Einheit nach einem der Ansprüche 2 oder 3,
dadurch gekennzeichnet,
dass die Fehlererkennungseinrichtung (5, 6, 10, 11, 14) den beiden Ausführungseinheiten ausgangsseitig nachgeschaltete Vergleichseinheit (14) enthält, die von den Ausführungseinheiten (1, 2) berechnete Ergebnisdaten und/oder deren Fehlerkorrekturcodierung nach Maßgabe einer Fehlerbehandlungsroutine auf das Vorhandensein eines Fehlers vergleicht und bei Vorhandensein eines Fehlers ein Fehlersignal ausgibt.
- 30 5. Programmgesteuerte Einheit nach einem der Ansprüche 2 bis 4,
dadurch gekennzeichnet,
- 35

5 dass die Fehlererkennungseinrichtung (5, 6, 10, 11, 14) zumindest eine zweite, zumindest einer Ausführungseinheit (1, 2) eingangsseitig vorgeschaltete Vergleichseinheit enthält, die einer jeweiligen Ausführungseinheit eingangsseitig zugeführte Eingangsdaten mit den mit einer Prüfsumme (z.B. Parity, CRC, ECC) versehenen Eingangsdaten nach Maßgabe einer Fehlerdetektierreoutine auf das Vorhandensein eines Fehlers vergleicht und bei Vorhandensein eines Fehlers ein Fehlersignal ausgibt.

- 10 6. Programmgesteuerte Einheit nach einem der vorstehenden Ansprüche,
dadurch gekennzeichnet,
dass zumindest ein Datenregister (8, 9) vorgesehen ist, welches zumindest einer der Ausführungseinheiten (1, 2) zugeordnet ist, welches ausgangsseitig sowohl mit den Eingängen der Ausführungseinheiten (1, 2) und den diesen vorgeschalteten Vergleichseinheit (5, 6) verbunden ist und in welchem die Eingangsdaten für die Ausführungseinheiten (1, 2) ablegbar sind.
- 15 7. Programmgesteuerte Einheit nach einem der vorstehenden Ansprüche,
dadurch gekennzeichnet,
dass ein Schattenregister vorgesehen ist, in dem die den Ausführungseinheiten (1, 2) vor der Berechnung in den Ausführungseinheiten zuletzt zugeführten Eingangsdaten abgelegt sind.
- 20 8. Programmgesteuerte Einheit nach Anspruch 7,
dadurch gekennzeichnet,
dass das Schattenregister als FIFO ausgebildet ist.
- 25 9. Programmgesteuerte Einheit nach einem der vorstehenden Ansprüche,
dadurch gekennzeichnet,
dass eine eingangsseitig mit der Fehlererkennungseinrichtung (5, 6, 10, 11, 14) und ausgangsseitig mit dem Schattenregister gekoppelte Steuereinrichtung vorgesehen ist, die ein Freigabesignal erzeugt und damit das Schattenregister erst dann freigibt, wenn von der Fehlererkennungseinrichtung (5, 6, 10, 11, 14) kein Fehler erkannt wird.
- 30 10. Programmgesteuerte Einheit nach einem der vorstehenden Ansprüche,
dadurch gekennzeichnet,
dass die programmgesteuerte Einheit als Mikrocontroller oder Mikroprozessor ausgebildet ist.
- 35 11. Verfahren zum Bertreiben einer programmgesteuerten Einheit nach einem der vorstehenden Ansprüche,
dadurch gekennzeichnet,

dass die Eingangsdaten und/oder die berechneten Ergebnisdaten und/oder deren Fehlercodierung miteinander verglichen werden und bei Nichtübereinstimmung des Ergebnisses des Vergleichs ein Fehlersignal erzeugt wird.

- 5 12. Verfahren nach Anspruch 11,
dadurch gekennzeichnet,
dass für jede Fehlerart ein eigenes Fehlersignal ausgegeben wird.
- 10 13. Verfahren nach einem der Ansprüche 11 bis 12,
dadurch gekennzeichnet,
dass die Eingangsdaten zunächst beiden Ausführungseinheiten (1, 2) zugeführt werden und im Anschluß daran aus den Eingangsdaten der Fehlerkorrekturcode gebildet wird.
- 15 14. Verfahren nach einem der Ansprüche 11 bis 13,
dadurch gekennzeichnet,
dass die abgelegten Eingangsdaten der letzten Berechnung erst wieder neu beschrieben werden, wenn ein Vergleich dieser Eingangsdaten oder der aus diesen Eingangsdaten berechneten Ergebnisdaten zu keinem Fehlersignal führt.
- 20 15. Verfahren nach einem der Ansprüche 11 bis 14,
dadurch gekennzeichnet,
dass die Ergebnisdaten erst bei Nichtvorhandensein eines Fehlersignals auf den Bus gelegt werden.

ROBERT BOSCH GMBH, 70442 Stuttgart

Programmgesteuerte Einheit und Verfahren

5

ZUSAMMENFASSUNG

Die vorliegende Erfindung schafft eine programmgesteuerte Einheit, mit einem einzigen Kontrollern (Core), der eine erste und zumindest eine zweite Ausführungseinheit aufweist, die in einem ersten Betriebsmodus unabhängig voneinander betreibbar sind und die in einem zweiten Betriebsmodus parallel die gleichen Befehle abarbeiten.

10

Figur 1)

Fig. 1

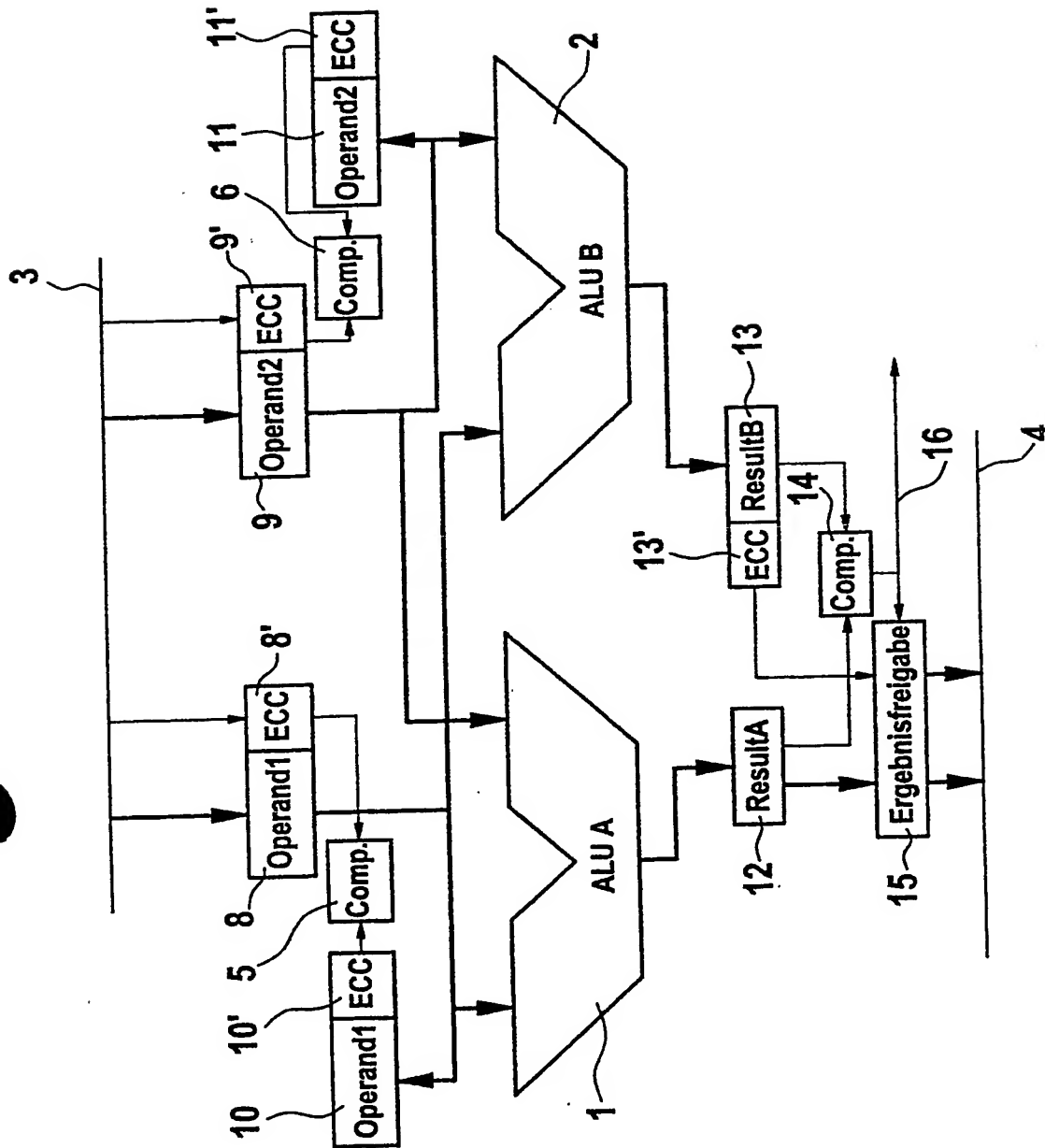


Fig. 2

